AMENDMENTS TO THE CLAIMS

The following listing of claims will replace all prior versions and listings of claims in the application.

Listing Of Claims

1. (Original) A daisy-chained device-mirroring architecture comprising:

a storage node N configured to mirror data on an entity representing node N-1;

a storage node N+1 daisy-chain-coupled via a first main link to, and configured to mirror data on, the node N; and

a storage node N+2 daisy-chain-coupled via a second main link to, and configured to mirror data on, the node N+1;

each of the nodes N and N+1 being operable under normal circumstances to forward downstream writes received thereby to the nodes N+1 and N+2 via the first and second main links, respectively;

the node N being operable to track acknowledgments by the node N+1 and by the node N+2 of writes that the node N has forwarded.

2. (Original) The architecture of claim 1, further comprising:

a standby node-shunting link (NSL) directly coupling the node N to the node N+2 so as to make the architecture tolerable of a single point of failure (SPOF) at node N+1.

3. (Original) The architecture of claim 2, wherein:

the node N, under circumstances of an SPOF at node N+1, is operable to forward writes directly to node N+2 via the NSL.

4. (Original) The architecture of claim 3, wherein:

the node N+2, under the circumstances of the SPOF at node N+1, is operable to generate a bitmap of writes received directly via the NSL from the node N.

5. (Original) The architecture of claim 4, wherein:

the node N+2, under the circumstances of the SPOF at node N+1, is operable to participate in reinstating the now-failed node N+1 by making a delta copy to the now-failed node N+1 of writes made since the SPOF occurred, based upon the bitmap.

6. (Original) The architecture of claim 1, further comprising:

a storage node N+3 daisy-chain-coupled via a third main link to, and configured to mirror data on, the node N+2;

the node N+2 being operable under normal circumstances to forward writes received thereby to the node N+3 via the third main link; and

the node N+1 being operable to track acknowledgments by the node N+2 and by the node N+3 of writes that the node N+2 has forwarded.

7. (Original) The architecture of claim 6, further comprising:

a standby node-shunting link (NSL) directly coupling the node N+1 to the node N+3 so as to make the architecture tolerable of a single point of failure (SPOF) at node N+2.

8. (Original) The architecture of claim 7, wherein:

the node N+1, under circumstances of an SPOF at node N+2, is operable to forward writes directly to node N+3 via the NSL.

9. (Original) The architecture of claim 8, wherein:

the node N+3, under the circumstances of the SPOF at node N+2, is operable to generate a bitmap of writes received directly via the NSL from the node N+1.

10. (Original) The architecture of claim 9, wherein:

the node N+3, under the circumstances of the SPOF at node N+2, is operable to participate in reinstating the now-failed node N+2 by making a delta copy to the now-failed node N+2 of writes made since the SPOF occurred, based upon the bitmap.

11. (Original) The architecture of claim 8, wherein:

the NSL is a second NSL;

the architecture further comprises

a first NSL directly coupling the node N to the node N+2 so as to make the architecture also tolerable of an SPOF at node N+1; and

the architecture is organized into overlapping triple units, each triple unit being defined by one of the NSLs.

12. (Original) The architecture of claim 11, wherein:

the first NSL defines a first triple unit as including the nodes N, N+1 and N+2; and

the second NSL defines a second triple unit as including the nodes N+1, N+2 and N+3.

13. (Original) The architecture of claim 8, wherein:

the node N+2 is operable to forward upstream to the node N+1 write-acknowledgments received from the node N+3.

14. (Original) The architecture of claim 1, wherein:

the node N+1 is operable to forward upstream to the node N write-acknowledgments acknowledgments received from the node N+2.

15. (Original) The architecture of claim 1, wherein:

storage node N is host-writable;

storage node N+1, under normal circumstances, is operable as a synchronous mirror of storage node N;

storage node N+2, under normal circumstances, is operable as an asynchronous mirror of storage node N; and

storage node N+2, under circumstances of a single point of failure (SPOF) at node N+1, is operable instead as a synchronous mirror of storage node N.

16. (Original) A method of operating a daisy-chained device-mirroring architecture that includes a storage node N configured to mirror data on an entity representing node N-1, a storage node N+1 daisy-chain-coupled via a first main link to, and configured to mirror data on, the node N, and a storage node N+2 daisy-chain-coupled via a second main link to, and configured to mirror data on, the node N+1, where each of the nodes N and N+1 are operable under normal circumstances to forward writes received thereby to the nodes N+1 and N+2 via the first and second main links, respectively, the method comprising:

tracking, at the node N, acknowledgments by the node N+1 and by the node N+2 of writes that the node N has forwarded.

17. (Original) The method of claim 16, further comprising:

forwarding from node N, under circumstances of a single point of failure (SPOF) at node N+1, writes directly to node N+2 via a standby node-shunting link (NSL) so as to make the architecture tolerable of an SPOF at node N+1.

18. (Original) The method of claim 16, further comprising:

generating, at the node N+2 under the circumstances of the SPOF at node N+1, a bitmap of writes received directly via the NSL from the node N.

19. (Original) The method of claim 18, further comprising:

participating, under the circumstances of the SPOF at node N+1, in a reinstatement of the now-failed node N+1 by causing the node N+2 to make a delta copy to the now-failed node N+1 of writes made since the SPOF occurred, based upon the bitmap.

20. (Original) The method of claim 16, wherein:

the architecture further includes a storage node N+3 daisy-chain-coupled via a third main link to, and configured to mirror data on, the node N+2, the node N+3 being operable to forward under normal circumstances to forward writes received thereby to the node N+3 via the third main link; and

the method further comprises:

tracking, at the node N+1, acknowledgements by the node N+2 and by the node N+3 of writes that the node N+2 has forwarded.

21. (Original) The method of claim 20, further comprising:

forwarding from node N+1, under circumstances of a single point of failure (SPOF) at node N+2, writes directly to node N+3 via a standby node-shunting link (NSL) so as to make the architecture tolerable of an SPOF at node N+2.

22. (Original) The method of claim 21, further comprising:

generating, at the node N+3 under the circumstances of the SPOF at node N+2, a bitmap of writes received directly via the NSL from the node N+1.

23. (Original) The method of claim 22, further comprising:

participating, under the circumstances of the SPOF at node N+2, in a reinstatement of the now-failed node N+2 by causing the node N+3 to make a delta copy to the now-failed node N+2 of writes made since the SPOF occurred, based upon the bitmap.

24. (Original) The method of claim 20, wherein:

forwarding upstream, from the node N+2, to the node N+1 write-acknowledgments received from the node N+3.

25. (Original) The method of claim 16, wherein:

forwarding upstream, from the node N+1 the node N, write-acknowledgments received from the node N+2.

26. (Original) The method of claim 16, wherein:

storage node N is host-writable;

operating storage node N+1, under normal circumstances, as a synchronous mirror of storage node N;

operating storage node N+2, under normal circumstances, as an asynchronous mirror of storage node N; and

operating storage node N+2, under circumstances of a single point of failure (SPOF) at node N+1, instead as a synchronous mirror of storage node N.

27. (Original) A method of operating a data storage architecture, the data storage architecture having a primary storage node, a secondary storage node, and a tertiary storage node, wherein the primary storage node is in communication with the secondary storage node, and the secondary storage node is in communication with the tertiary storage node, each of the storage nodes employing a sidefile and bitmap, the method comprising:

utilizing a sidefile of the primary storage node to track receipt of write acknowledgements sent by each of the secondary storage node and the tertiary storage node.

28. (Original) The method according to claim 27, further comprising:

providing the sidefile with at least first and second logic fields; and

wherein the utilizing of the sidefile includes

recording receipt of write-acknowledgements from the secondary storage node and the tertiary storage node in the first and second logic fields, respectively.

29. (Currently Amended) A machine-readable medium including instructions execution of which by a machine causes operation of a data storage architecture, the data storage architecture

having a primary storage node, a secondary storage node, and a tertiary storage node, wherein the primary storage node is in communication with the secondary storage node~~,~~ and the secondary storage node is in communication with the tertiary storage node, ~~and the tertiary storage node is in communication with the quaternary storage node,~~ each of the storage nodes employing a sidefile and bitmap, the machine-readable instructions comprising:

a code segment that utilizes a sidefile of the primary storage node to track receipt of write acknowledgements sent by each of the secondary storage node and the tertiary storage node.

30. (Original) A primary storage node in a data storage architecture, the primary storage node being in communication with a secondary storage node, the secondary storage node being in communication with a tertiary storage node; the primary storage node comprising:

a memory; and

a processing unit to configure the memory;

a least a portion the memory being configured as a sidefile that includes at least first and second logic fields in which receipt of write-acknowledgements from the secondary storage node and the tertiary storage node are recordable in the first and second logic fields, respectively.

31. (Original) A daisy-chained device-mirroring architecture comprising:

a storage node N configured to mirror data on an entity representing node N-1;

a storage node N+1 daisy-chain-coupled via a first main link to, and configured to mirror data on, the node N, and

a storage node N+2 daisy-chain-coupled via a second main link to, and configured to mirror data on, the node N+1;

the node N+1 being operable to forward upstream write-acknowledgements received thereby.

32. (Original) The architecture of claim 1, further comprising:

a storage node N+3 daisy-chain-coupled via a third main link to, and configured to mirror data on, the node N+2;

wherein the node N+2 is operable to forward upstream write-acknowledgements received thereby.

33. (Original) A method of operating a daisy-chained device-mirroring architecture that includes a storage node N configured to mirror data on an entity representing node N-1, a storage node N+1 daisy-chain-coupled via a first main link to, and configured to mirror data on, the node N, and a storage node N+2 daisy-chain-coupled via a second main link to, and configured to mirror data on, the node N+1, where each of the nodes N and N+1 are operable under normal circumstances to forward writes received thereby to the nodes N+1 and N+2 via the first and second main links, respectively, the method comprising:

forwarding upstream write-acknowledgements received at the node N+1.

34. (Original) The method of claim 33, wherein:

the architecture further includes a storage node N+3 daisy-chain-coupled via a third main link to, and configured to mirror data on, the node N+2; and

the method further comprises forwarding upstream write-acknowledgements received at the node N+2.

<remainder of page intentionally left blank>